

TELEROBOTIC WORKSTATION DESIGN AID

K. Corker, E. Hudlicka, D. Young, N. Cramer

BBN System & Technologies Corporation
Cambridge, Massachusetts 02019

1.0 Introduction

Telerobot systems are being developed to support a number of space mission applications. In low earth orbit, telerobots and teleoperated manipulators will be used in shuttle operations and space station construction/maintenance. Free flying telerobotic service vehicles will be used at low and geosynchronous orbital operations. Rovers and autonomous vehicles will be equipped with telerobotic devices in planetary exploration.

In all of these systems human operators will interact with the robot system at varied levels during the scheduled operations. The human operators may be in either orbital or ground-based control systems.

In order to assure integrated system development and maximum utility across these systems, designers must be sensitive to the constraints and capabilities that the human brings to system operation and must be assisted in applying these "Human Factors" to system development. The simulation and analysis system which is the topic of this paper is intended to serve the needs of system analysis/designers as an integrated workstation in support of telerobotic design.

1.1. Design and Development in Space Telerobotic Systems

The design environment in space telerobotic systems imposes several unique constraints on the development of a design aid. This is especially true for the inclusion of human operators in system control. The telerobot system serves as an extension of human perceptual, cognitive, and manipulative abilities to a remote site of operation. The design process, then, should be responsive to human interface requirements. However, there are a number of constraints in the telerobot design process that make inclusion of operator-focussed concerns difficult to implement. Taking the development of the Telerobot Testbed at the Jet Propulsion Laboratory as characteristic of the difficulties in this large-scale human/system design, we find the following.

This work has been supported on a cooperative basis by Jet Propulsion Laboratory and NASA-Ames Research Center Contract #200-216-55

- (1) **Multiple Subsystems:** Independently developed, but interactive in operation the telerobot subsystems (AI Planner, run-time controller, manipulator control module, sensing and perception, executive monitor) offer a challenging amount of information to a human operator/monitor. All of this information is to be channelled through the operator control station. It is this control station design that is the focus of our design aiding tool.
- (2) **Variable Development Cycles:** Given the independent development process, the level to which subsystem function can be specified varies among the modules. This means the information provided to and action required of the operator varies in its specificity
- (3) **Evolving Mission Requirements:** As decisions are made as to the extent of telerobot servicing required, or the level of operator intervention in that operation, the tasks and performance criteria for system operation change. The boundaries of the operator tasks profile are a moving target for interface design to capture.
- (4) **Undetermined Manning and Training Requirements:** Because of the evolving nature of the mission and system requirements the number of operation required and whatever special training then may require is not yet specified.

In order to deal with such a fluid design environment, an aiding tool must provide the designer/analyst with a flexible, modular and modifiable system that takes into account the nature of the human operations, the nature of the operator interface, description of the mission, the constraints and capabilities of the equipment to perform the operations and some performance assessment process to judge the relative value of one design versus another.

1.2 Issues Evaluated In Space Telerobotic Design

Before describing our evaluation tool, we will briefly mention the particular issues we are attempting to resolve in relation to the JPL Telerobot Testbed.

- **Control Station Layout/Design:** The Operator Control Station (OCS) is the single point of interface between the operator(s) and the telerobot control subsystems, the robot arms, the satellite and the sensing and perception system. The operator will use the OCS for monitoring, control, and diagnosis of mission progress. Our design aid is intended to assist in determining: What information is needed to support the operator, what control mechanisms should be provided, how should this information be provided, and what is the optimal configuration of the OCS for operator use.
- **Automatic vs. Manual Telerobotic Control:** Given that the telerobot system has the capacity for control by computer or transmission of human control commands to the robot arms, there is an issue of allocation of control authority between the human and the AI Planner (AIP) subsystem. The design aid will allow system designer to explore control strategies by making assignments of varied control responsibilities for parts of the mission and examining the results of those strategies on performance and operator load. Control can be designated to reside with the operator, the AIP, or to be traded between them on portions of the task, or to be shared in task cooperative control.

- **Management of the Telerobotic Systems:** The coordination and diagnosis of subsystem operation is a critical task in the Telerobot Testbed. The design aid will provide the ability to allow the operator to view varied messages and message traffic among subsystems. Designers can induce simulated errors and explore diagnostic and error recovery strategies.
- **Manning and Training Issues:** The operational requirements for the telerobot system do not yet specify one or two operators, or suggest the operational affects of earth-based versus orbital operation. The design aid will support investigation of performance with one or two operators and allow for exploration of the impact of time delay or bandwidth limitations on feedback or control information.

2.0 Design Aid

We have taken the approach of providing telerobot designer and analysts with a simulation tool. The basic architecture of that tool is illustrated in Figure 1. The architecture is illustrative of the modular nature of the overall aiding system implementation. We will describe that implementation, and then detail the function of the major components as applied to the Telerobot Testbed.

TELEROBOTIC EMULATION MODEL ARCHITECTURE

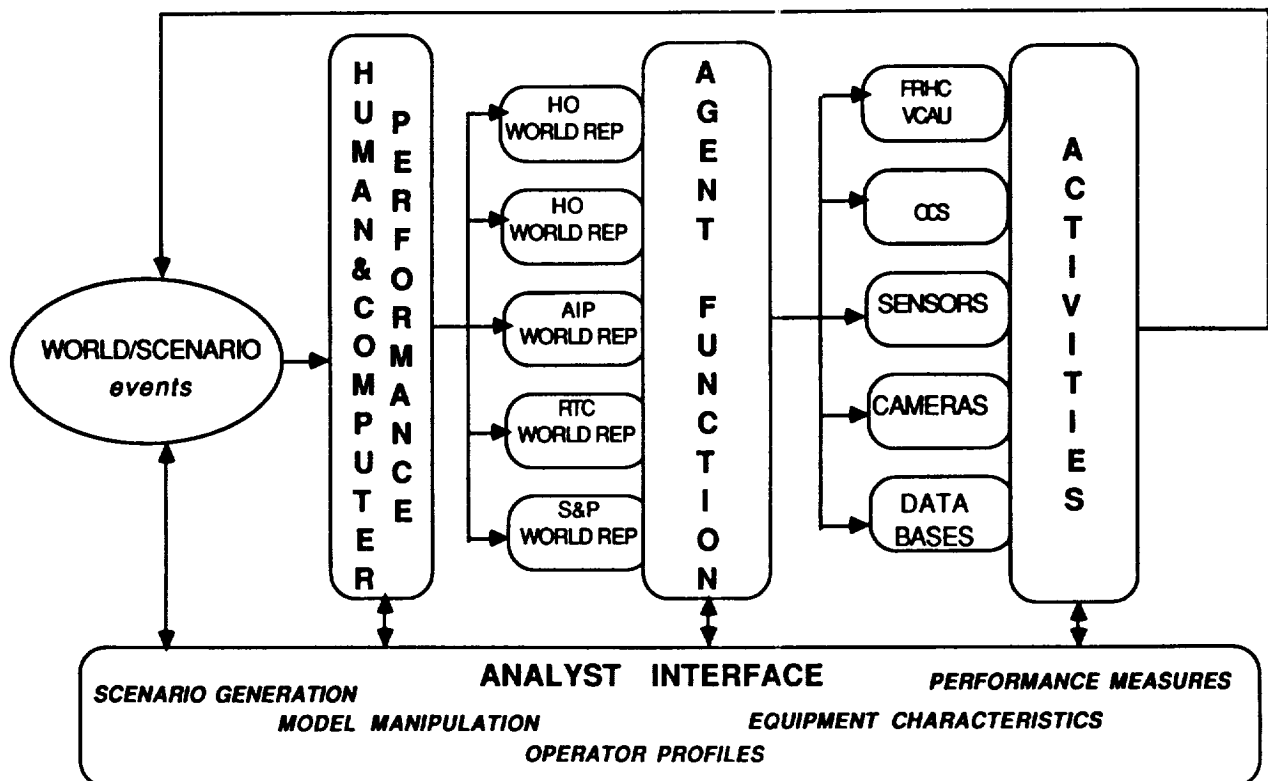


Figure 1

2.1 System Implementation

The system is implemented in Zeta Lisp in the Symbolics LISP machine environment. It uses the FLAVORTM object system. As a Common Lisp specification for object-oriented programming develops, the system will be converted to Common Lisp and its associated object language. Common Lisp is expected to become the shared language of the artificial intelligence and LISP communities; this conversion will make this system compatible with a wide range of other systems.

The aiding system has three components:

- a simulation driver,
- libraries of object descriptions,
- and a library of input/output utilities,

The simulation driver allows a user to run the simulation at fixed (and user-definable) increments of time, or by letting the occurrence of events drive the clock, providing a discrete time, or discrete event simulation. The libraries of object descriptions include control station configuration and function, robot arms, sensors, and controls, human operator's control strategies, perceptual and motor characteristics, decision and diagnostic procedures, the task environment and tools, and a variety of other supporting object types. These will be detailed below. The input and output utilities include the ability to display the position and status of the mobile objects in the system, to display the procedural hierarchy of activities, to display inter subsystem communication, and to display the level and type of loading on the operator.

2.2 Functional Modules

Scenario: In order to give the designer a tool to exercise the telerobot system and operator models in a variety of operations, we have created a module of object code that contains operational procedures. These procedures are examinable and modifiable using screen-based tools.

We have selected a portion of the telerobotic demonstration to exercise our simulation tools. Beginning with the "flat" sequence of planned activities and communication among the system models that was used to encode the telerobot scenario, we have constructed a procedural hierarchy of goal-oriented activities organized from the point of view of the "cognitive agents" in the simulation, i.e., the AI planner and the human operator.

The scenario operation is represented by a semi-lattice of goals, subgoals, and activities and procedures for meeting those goals. (Figure 2). This representation provides an abstraction of the scenario activities. The abstraction provides the designer/analyst with an overall view of allowable and selected paths to the desired sequences of activities to accomplish the demonstration tasks. It also provides a convenient tool for alternative plan construction and task-level diagnostic activity. The procedural representation, the logical and procedural constraints that determine allowable actions will be made available as a mouse-sensitive object display. This display will show the current sequences of simulator action. When the activity nodes are selected a more detailed description of that activity will be provided. The details of an activity such as its duration, or preconditions, or termination requirements, tolerance, or agent required to perform that activity are all modifiable by the designer.

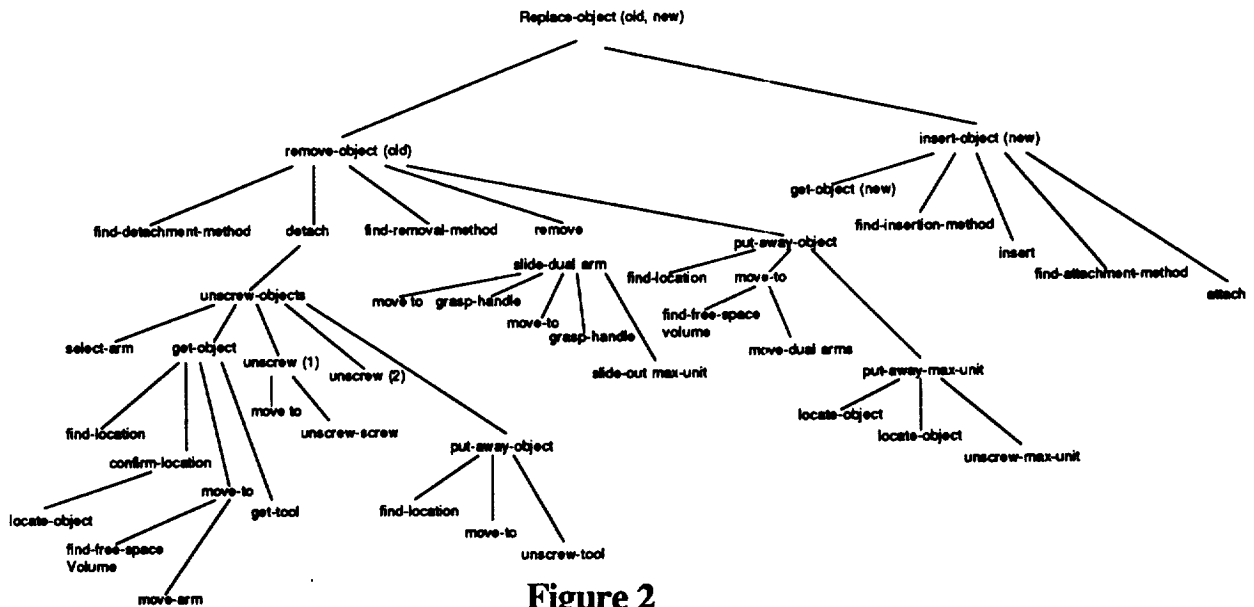


Figure 2

Human Operator Models: The framework in which the individual models of human performance are embedded is the "operator object". The operator object interacts with the telerobotic demonstration through perceptual processes and activities. The operator has an individually defined "updatable world representation". This world representation is a description of the world as the operator knows it. It contains rules for decision, and an awareness of external events as they are passed through the operator's sensory systems. Each of the slots of the operator object's world state contain information as to the source of the information in the slot. For example, an alert state will contain a slot identifying the source of the information about that alert, e.g., "flashing warning, position X, Y, on the control station." The world representation in the object is a decaying store of information. World state changes are entered into the representation with time signatures and priorities. The frequency with which state changes occur and their relative priorities, will determine the length of time that state information remains available for decisions and rule applicable. The world representation also contains a queue of action that should be taken at some time in the future, a "goal queue". The operator interacts with the task-world through sensory systems.

These sensory processes are modeled by our system. We have implemented a visual model.

Visual Scanning: Each of the activities in the simulation script has an attribute which identifies what equipment (and what sequence of interaction) it requires. The operator(s) of the telerobotic system must constantly "scan" their equipment and displays to keep high resolution (foveal) information updated. In order to account for the time and movement required to find and fix target data in the telerobotic tasks, we are implementing a model of visual scanning in two stages.

The first "active gaze" represents the focused and directed movement from the current point of regard to a target point. The action is characteristic of actions in which the to be attended object is in a known position. The motion is a straight line from the present position to the target (though there may be a contribution through head movement, we will not consider that at this time.) The operator is assumed to be 18 in. from the center of the OCS. The velocity of motion is 100 degrees per second and the foveal fixation area is .5 degrees. There is a 200 msec. pause between eye motions (i.e., saccades).

The second type of gaze is a monitoring or search pattern. The saccades in in such a search pattern typically last for 50 msec and cover about 10 degrees. Again there is a 200 msec pause between movements. The effective radius of a fixation in this scan is about 14 degrees around the center of fixation.

In order for an operator to "notice" any information or to make judgments to guide teleoperation, the operator object must have scanned the relevant source of visual information. If the operator's visual attention is focused on a particular display or control he/she will not respond to other information until it is viewed.

This model provides information about visual latencies and the positioning of displays on the OCS. We hope to implement more sophisticated visual model to include depth perception and visual acuity issues for display design. We have also implemented a decision-making process.

Decision Making and Diagnosis: The operator object's updatable world representation contains object descriptions of those actions and events on which decisions need to be made. Using a rule-based decision framework, the operator object applies rules to the world representation to determine what action is required. If the rules are insufficient to make a decision (and an activity selection is required) then the operator object will look at the full procedure hierarchy to select alternative and appropriate actions to follow. The approximate time required to make a decision or to pursue a diagnostic path is calculated and that time is added to the simulation performance time.

The idea of the human operator as a limited capacity channel in complex system operation is central to our model. Designers should be aware of the attentional and workload implication of a given design. In order to give some insight into the loading on the operator of a given operation we have implemented a performance capacity model for the operator.

Task and Operator Loads: As previously described, tasks are represented in a hierarchy of procedure. At each level of that hierarchy an assessment is made of the load associated with that activities performance. These assessments are currently estimates, but future activity descriptions will have estimates provided by expert's subjective opinion and represented on a scale from 0-7 with 7 indicating complete or full load. These task/activity loads are further refined by being divided according to resources that an operator can bring to bear to perform the task. We have partitioned those resources to be visual, auditory (which includes both speech and hearing), cognitive and psychomotor (VACP). In addition, we are implementing a moderating or gain function, which we denote as attention, on the use of those resources. The process by which the task demands are met by the operator resources is mediated by attention.

Efficient matching of resources to demands, i.e., resources applied equal to and for a duration commensurate with demands, can be expected of a well-trained and unstressed operator. In appropriate allocation of resources is the result of lack of training or a change in the attention allocation function.

Agent Functions: Agent functions stand as links between models are going to carryout. The agent functions to free the designer from the assumption that a particular activity is to be performed by one or more human operators or by an intelligent machine. For example, communication (as an agent function) knows that in order to communicate there is a source and a destination for information and a message to be passed. If the communication is among submodules of the telerobot system then one type of message protocol and one type or resource is involved, i.e., a network transactions protocol, and bus or fibernet links. If human operators are communicating then auditory models are invoked. Human communication with the system takes place through the OCS. Figure 3 illustrates the role of agents linking operators to activities.

Human Operator Object Relation to Activities

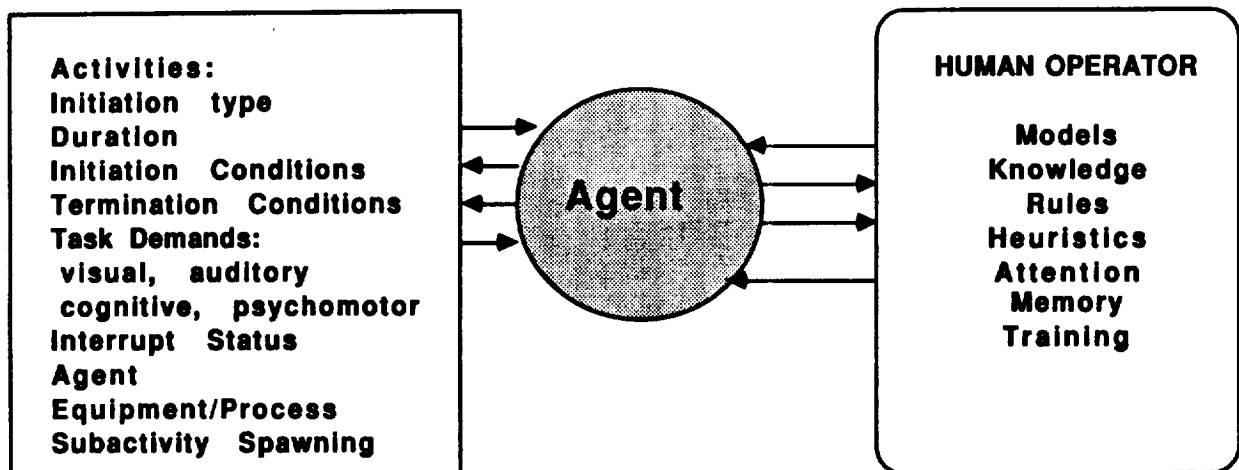


Figure 3

Equipment: We have focussed on the OCS design in our simulation. Figure 4.

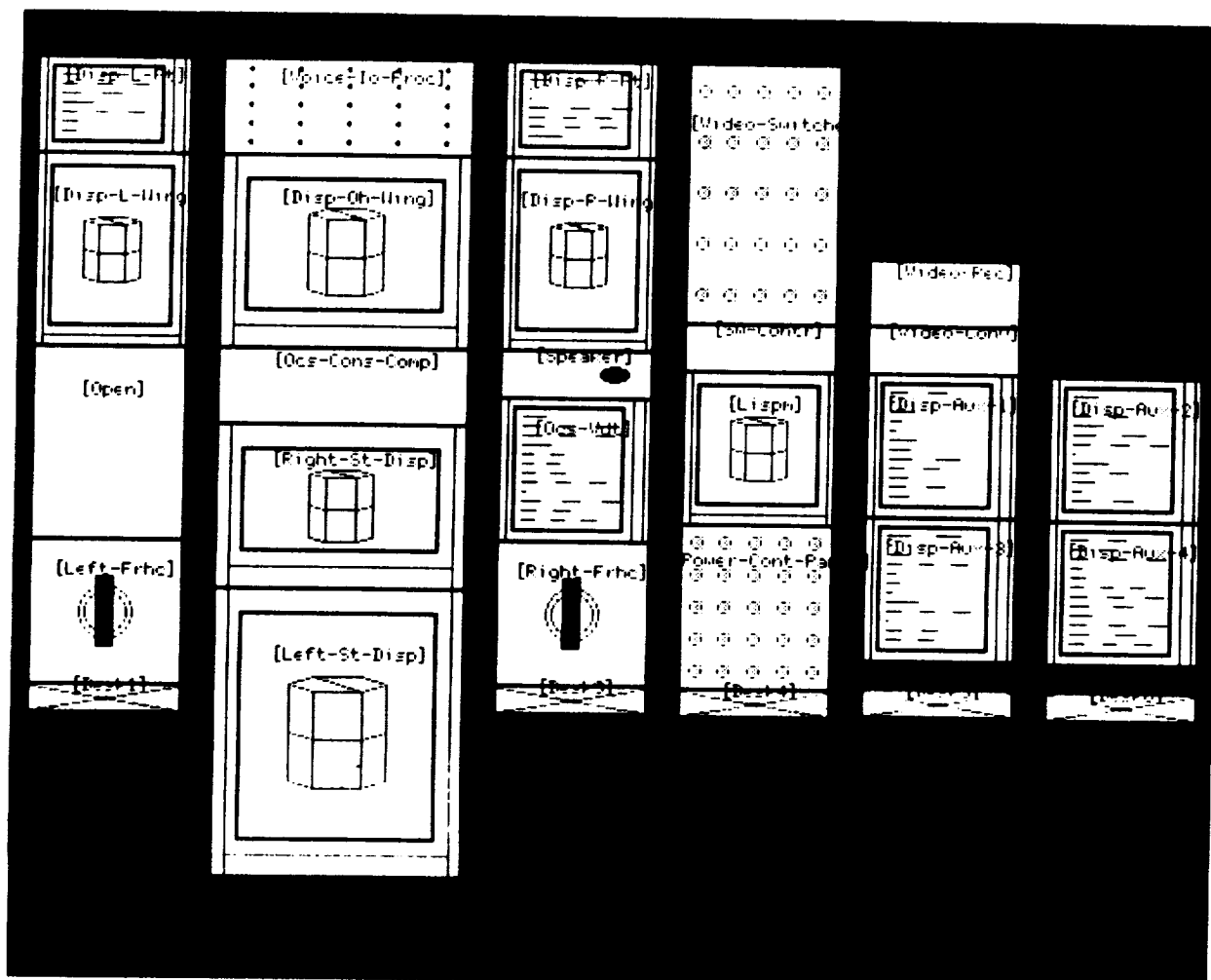


Figure 4

Illustrates an object-based control panel. The switches, controls and displays are objects whose function moves if the placement of the device is changed. All of the control and display functions are "Mouse Sensitive" and can be selected and moved to reconfigure the layout of the control station. The switches and controls can be made "Active" in the mousing them initiates the appropriate function in the software simulation. The panel is, therefore, a reconfigurable display the control. The information or control that is relevant at a particular moment in the simulation is also highlighted as the sequence of mission operations occur. Designers can reconfigure the control

station, assume different types of display, or levels of automation and then run the simulation with the candidate configuration. The visual model and workman will respond to a given configuration with a unique profile of operator activity. We also provide a view of the ongoing robotic task. Figure 5. We also represent communication among the subsystem components. Figure 6. This display is screen assessable so that the design can track decision and control among components.

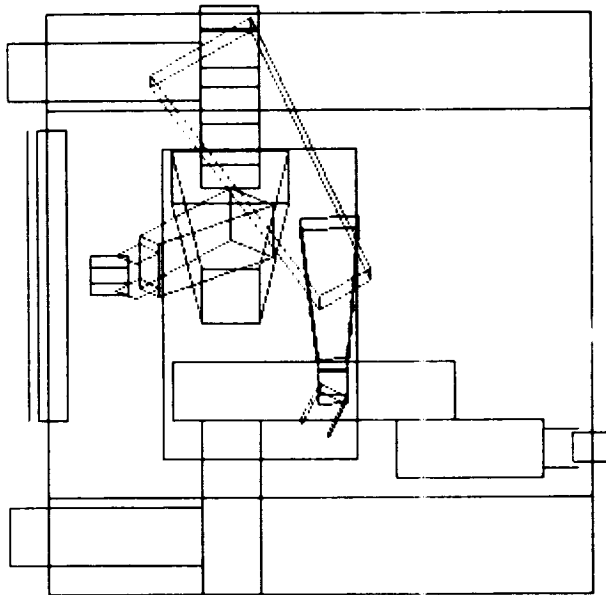


Figure 5

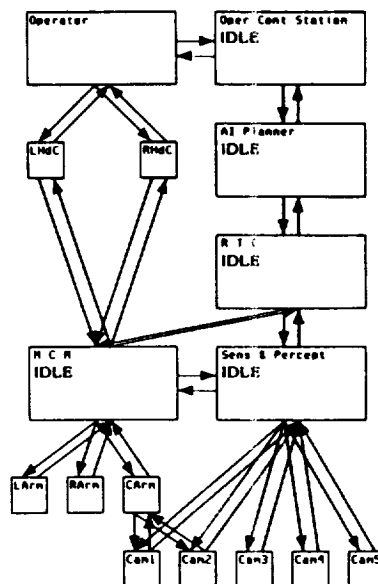


Figure 6

Activities: Each activity has operations that describe its function at each time increment ("tick") of the simulation, when the activity is initialized, and when the activity is terminated. For example, for activities with sub-activities, the tick operation is responsible for starting a sub-activity, waiting until it has finished executing, and then starting the next sub-activity. When an activity is created, its tick operator performs the instructions described in the activity's initialization procedures variable, and when an activity is finished executing its termination procedures are performed.

FLAVORS™ can be defined by combining several other FLAVORSTM and adding new operations or state variables. In this way, base activity types are created to support activities which can perform their sub-activities:

- either sequentially or in parallel,
- allowing interruption or not,
- with a fixed or variable execution time,
- allowing further subactivity spawning or not.

By making instances of activity FLAVORSTM and assigning them to the appropriate active-objects and agent, the crew-member/agent/can "tick" their activities and take further actions based on the state(s) of their current actions.

Activities themselves are organized in a hierarchic structure. An activity has a parent that spawned it, a list of children that it has spawned, a procedure to carry out when it receives an act message, and a set of procedures to carry out when it is created, terminated, interrupted, suspended, and/or aborted.

Activities can be performed by one or several agents. The evaluation system requires no assumption as to the particular technology used in performing an action, nor does it require rigid specification of the results of action. The message-passing protocol establishes a modular discipline on the development of the simulation environment. This allows, for example, an investigation of performance assuming an automatic control for a task compared with the same task performed in a teleoperated mode. The changes in system response are propagated through the simulation/evaluation as a function of the message passing protocols associated with the object-oriented code. The demonstration assumes that an specific "active object" will normally perform a given activity. An active object is defined as an object that has a list of activities that it is carrying out. When an agent is active (i.e., when it has received an appropriate message) it activates each of its activities and they proceed to carry out their function. This type of object, the active object agent" is a component of many other types of objects, including telerobots, AI Planner, human operators and etc.

The object-oriented, model-based simulation described provides the designer of telerobotic systems and operations a tool to investigate the feasibility and performance impact of various mixes of human operator and autonomous control in space telerobotics. The system produces efficient and effective methods for early identification of procedural bottlenecks and control conflict. Iterative refinement of the design of control architecture can focus the designer's attention and resources on particularly promising or troublesome interactions. Once identified these critical interaction points can be explored in part-task simulation. Additionally, the system provides a preliminary prediction of operator performance given a particular control station configuration. Issues of control station usability can then be addressed prior to commitment to a prototype configuration. Finally, the system design tool serves as a repository of data as performance and prototyping experiments are performed on telerobotic component modules, and the design is incrementally decided upon.



